

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S39	73	(history with (table list directory register buffer queue\$4 memory)) same ((invalidat\$3 in-validat\$3) with (entry line data address))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 15:36
S40	3	((re-validat\$3 revalidat\$3) same (entry line data address)) and S39	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 15:43
S41	744	(history with (table list directory register buffer queue\$4 memory)) and ((invalidat\$3 in-validat\$3) with (entry line data address))	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 15:37
S42	3	((re-validat\$3 revalidat\$3) and (entry line data address)) and S39	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 15:37
S43	16	((re-validat\$3 revalidat\$3) and (entry line data address)) and S41	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 15:42
S44	7018	711/118-146.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 15:43
S45	26	((re-validat\$3 revalidat\$3) same (entry line data address)) and S44	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 16:07
S46	28	"5197144".uref.	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 16:07
S47	1	((re-validat\$3 revalidat\$3) same (entry line data address)) and S46	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 16:13
S48	1	((re-validat\$3 revalidat\$3) and (entry line data address)) and S46	US-PGPUB; USPAT; EPO; JPO; DERWENT	OR	ON	2005/07/07 16:14

PORTAL
USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login
Search: The ACM Digital Library The Guide
+invalidate +history +cache +memory +controller +dirty entr **SEARCH**

Feedback Report a problem Satisfaction survey

Terms used

invalidate history cache memory controller dirty entry line data address

Found 30 of 157,956

Sort results by

relevance Save results to a BinderTry an Advanced Search

Display results

expanded form Search TipsTry this search in The ACM Guide Open results in a new window

Results 1 - 20 of 30

Result page: 1 2 [next](#)Relevance scale **1 Cache Memories**

Alan Jay Smith

September 1982 **ACM Computing Surveys (CSUR)**, Volume 14 Issue 3Full text available:  pdf(4.61 MB)Additional Information: full citation, references, citations, index terms**2 Implementing a cache consistency protocol**

R. H. Katz, S. J. Eggers, D. A. Wood, C. L. Perkins, R. G. Sheldon

June 1985 **ACM SIGARCH Computer Architecture News, Proceedings of the 12th annual international symposium on Computer architecture**, Volume 13 Issue 3Full text available:  pdf(803.11 KB)Additional Information: full citation, citations, index terms

Keywords: ownership-based protocols, shared bus multicomputer cache consistency, single chip implementation, snooping caches

**3 Area efficient architectures for information integrity in cache memories**

Seongwoo Kim, Arun K. Soman

May 1999 **ACM SIGARCH Computer Architecture News, Proceedings of the 26th annual international symposium on Computer architecture**, Volume 27 Issue 2Full text available:  pdf(227.09 KB)Additional Information: full citation, abstract, references, citations, index terms Publisher Site

Information integrity in cache memories is a fundamental requirement for dependable computing. Conventional architectures for enhancing cache reliability using check codes make it difficult to trade between the level of data integrity and the chip area requirement. We focus on transient fault tolerance in primary cache memories and develop new architectural solutions, to maximize fault coverage when the budgeted silicon area is not sufficient for the conventional configuration of an error checki ...

**4 Tango: a hardware-based data prefetching technique for superscalar processors**

Shlomit S. Pinter, Adi Yoaz

December 1996 **Proceedings of the 29th annual ACM/IEEE international symposium on Microarchitecture**

Full text available:  pdf(1.46 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We present a new hardware-based data prefetching mechanism for enhancing instruction level parallelism and improving the performance of superscalar processors. The emphasis in our scheme is on the effective utilization of slack time and hardware resources not used for the main computation. The scheme suggests a new hardware construct, the program progress graph (PPG), as a simple extension to the branch target buffer (BTB). We use the PPG for implementing a fast pre-program counter pre-PC, that ...

Keywords: LRU mechanism, SPEC92 benchmark, Tango, base line architecture, branch target buffer, hardware resources, hardware-based data prefetching technique, instruction level parallelism, instruction prefetching, memory reference instructions, parallel processing, performance, program progress graph, simulation results, slack time, superscalar processors

5 Piranha: a scalable architecture based on single-chip multiprocessing

Luiz André Barroso, Kourosh Gharachorloo, Robert McNamara, Andreas Nowatzky, Shaz Qadeer, Barton Sano, Scott Smith, Robert Stets, Ben Verghese

May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture**, Volume 28 Issue 2

Full text available:  pdf(191.10 KB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The microprocessor industry is currently struggling with higher development costs and longer design times that arise from exceedingly complex processors that are pushing the limits of instruction-level parallelism. Meanwhile, such designs are especially ill suited for important commercial applications, such as on-line transaction processing (OLTP), which suffer from large memory stall times and exhibit little instruction-level parallelism. Given that commercial applications constitute by fa ...

6 Recovery protocols for shared memory database systems

Lory D. Molesky, Krithi Ramamirtham

May 1995 **ACM SIGMOD Record , Proceedings of the 1995 ACM SIGMOD international conference on Management of data**, Volume 24 Issue 2

Full text available:  pdf(1.65 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Significant performance advantages can be gained by implementing a database system on a cache-coherent shared memory multiprocessor. However, problems arise when failures occur. A single node (where a *node* refers to a processor/memory pair) crash may require a reboot of the entire shared memory system. Fortunately, shared memory multiprocessors that isolate individual node failures are currently being developed. Even with these, because of the side effects of the cache coherency protocol, ...

7 4.2BSD and 4.3BSD as examples of the UNIX system

John S. Quarterman, Abraham Silberschatz, James L. Peterson

December 1985 **ACM Computing Surveys (CSUR)**, Volume 17 Issue 4

Full text available:  pdf(4.07 MB)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

This paper presents an in-depth examination of the 4.2 Berkeley Software Distribution, Virtual VAX-11 Version (4.2BSD), which is a version of the UNIX Time-Sharing System. There are notes throughout on 4.3BSD, the forthcoming system from the University of California at Berkeley. We trace the historical development of the UNIX system from its conception in 1969 until today, and describe the design principles that have guided this development. We then present the internal data structures and ...

8 Verification techniques for cache coherence protocols

Fong Pong, Michel Dubois

March 1997 **ACM Computing Surveys (CSUR)**, Volume 29 Issue 1

Full text available:  [pdf\(1.25 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In this article we present a comprehensive survey of various approaches for the verification of cache coherence protocols based on state enumeration, (symbolic model checking, and symbolic state models. Since these techniques search the state space of the protocol exhaustively, the amount of memory required to manipulate that state information and the verification time grow very fast with the number of processors and the complexity of the protocol mechanism ...

Keywords: cache coherence, finite state machine, protocol verification, shared-memory multiprocessors, state representation and expansion

9 Owner prediction for accelerating cache-to-cache transfer misses in a cc-NUMA architecture

Manuel E. Acacio, José González, José M. García, José Duato

November 2002 **Proceedings of the 2002 ACM/IEEE conference on Supercomputing**

Full text available:  [pdf\(120.57 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Cache misses for which data must be obtained from a remote cache (cache-to-cache transfer misses) account for an important fraction of the total miss rate. Unfortunately, cc-NUMA designs put the access to the directory information into the critical path of 3-hop misses, which significantly penalizes them compared to SMP designs. This work studies the use of owner prediction as a means of providing cc-NUMA multiprocessors with a more efficient support for cache-to-cache transfer misses. Our propo ...

10 Don't use the page number, but a pointer to it

André Seznec

May 1996 **ACM SIGARCH Computer Architecture News , Proceedings of the 23rd annual international symposium on Computer architecture**, Volume 24 Issue 2

Full text available:  [pdf\(1.26 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Most newly announced high performance microprocessors support 64-bit virtual addresses and the width of physical addresses is also growing. As a result, the size of the address tags in the L1 cache is increasing. The impact of on chip area is particularly dramatic when small block sizes are used. At the same time, the performance of high performance microprocessors depends more and more on the accuracy of branch prediction and for reasons similar to those in the case of caches the size of the Br ...

Keywords: address width, indirect-tagged caches, reduced branch target buffers, tag implementation cost

11 Performance of database workloads on shared-memory systems with out-of-order processors

Parthasarathy Ranganathan, Kourosh Gharachorloo, Sarita V. Adve, Luiz André Barroso

October 1998 **Proceedings of the eighth international conference on Architectural support for programming languages and operating systems**, Volume 33 , 32 Issue 11 , 5

Full text available:  [pdf\(1.62 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Database applications such as online transaction processing (OLTP) and decision support systems (DSS) constitute the largest and fastest-growing segment of the market for multiprocessor servers. However, most current system designs have been optimized to perform well on scientific and engineering workloads. Given the radically different behavior of database workloads (especially OLTP), it is important to re-evaluate key system design decisions in the context of this important class of applicatio ...

12 Hardware-assisted replay of multiprocessor programs

David F. Bacon, Seth Copen Goldstein

December 1991 **ACM SIGPLAN Notices , Proceedings of the 1991 ACM/ONR workshop on Parallel and distributed debugging**, Volume 26 Issue 12

Full text available:  [pdf\(1.20 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

13 Using destination-set prediction to improve the latency/bandwidth tradeoff in shared-memory multiprocessors

Milo M. K. Martin, Pacia J. Harper, Daniel J. Sorin, Mark D. Hill, David A. Wood

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2

Full text available:  [pdf\(220.76 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

Destination-set prediction can improve the latency/bandwidth tradeoff in shared-memory multiprocessors. The destination set is the collection of processors that receive a particular coherence request. Snooping protocols send requests to the maximal destination set (i.e., all processors), reducing latency for cache-to-cache misses at the expense of increased traffic. Directory protocols send requests to the minimal destination set, reducing bandwidth at the expense of an indirection through the d ...

14 Compiler and hardware support for cache coherence in large-scale multiprocessors: design considerations and performance study

Lynn Choi, Pen-Chung Yew

May 1996 **ACM SIGARCH Computer Architecture News , Proceedings of the 23rd annual international symposium on Computer architecture**, Volume 24 Issue 2

Full text available:  [pdf\(1.48 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In this paper, we study a hardware-supported, compiler directed (HSCD) cache coherence scheme, which can be implemented on a large-scale multiprocessor using off-the-shelf microprocessors, such as the Cray T3D. It can be adapted to various cache organizations, including multi-word cache lines and byte-addressable architectures. Several system related issues, including critical sections, inter-thread communication, and task migration have also been addressed. The cost of the required hardware sup ...

15 Speculative synchronization: applying thread-level speculation to explicitly parallel applications

José F. Martínez, Josep Torrellas

October 2002 **Proceedings of the 10th international conference on Architectural support for programming languages and operating systems**, Volume 36 , 30 , 37 Issue 5 , 5 , 10

Full text available:  [pdf\(1.49 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

Barriers, locks, and flags are synchronizing operations widely used programmers and parallelizing compilers to produce race-free parallel programs. Often times, these operations are placed suboptimally, either because of conservative assumptions about the program, or merely for code simplicity. We propose *Speculative Synchronization*, which applies the philosophy behind Thread-Level Speculation (TLS) to explicitly parallel

applications. Speculative threads execute past active barriers, busy ...

16 Informing memory operations: memory performance feedback mechanisms and their applications

Mark Horowitz, Margaret Martonosi, Todd C. Mowry, Michael D. Smith

May 1998 **ACM Transactions on Computer Systems (TOCS)**, Volume 16 Issue 2

Full text available:  [pdf\(344.74 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Memory latency is an important bottleneck in system performance that cannot be adequately solved by hardware alone. Several promising software techniques have been shown to address this problem successfully in specific situations. However, the generality of these software approaches has been limited because current architectures do not provide a fine-grained, low-overhead mechanism for observing and reacting to memory behavior directly. To fill this need, this article proposes a new class ...

Keywords: cache miss notification, memory latency, processor architecture

17 A high-level abstraction of shared accesses

Peter J. Keleher

February 2000 **ACM Transactions on Computer Systems (TOCS)**, Volume 18 Issue 1

Full text available:  [pdf\(183.57 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)

We describe the design and use of the tape mechanism, a new high-level abstraction of accesses to shared data for software DSMs. Tapes consolidate and generalize a number of recent protocol optimizations, including update-based locks and recorded-replay barriers. Tapes are usually created by "recording" shared accesses. The resulting recordings can be used to anticipate future accesses by tailoring data movement to application semantics. Tapes-based mechanisms a ...

Keywords: DSM, programming libraries, shared memory, update protocols

18 Performance characterization of a Quad Pentium Pro SMP using OLTP workloads

Kimberly Keeton, David A. Patterson, Yong Qiang He, Roger C. Raphael, Walter E. Baker

April 1998 **ACM SIGARCH Computer Architecture News, Proceedings of the 25th annual international symposium on Computer architecture**, Volume 26 Issue 3

Full text available:   [pdf\(1.58 MB\)](#) [Publisher Site](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Commercial applications are an important, yet often overlooked, workload with significantly different characteristics from technical workloads. The potential impact of these differences is that computers optimized for technical workloads may not provide good performance for commercial applications, and these applications may not fully exploit advances in processor design. To evaluate these issues, we use hardware counters to measure architectural features of a four-processor Pentium Pro-based se ...

19 An effective on-chip preloading scheme to reduce data access penalty

Jean-Loup Baer, Tien-Fu Chen

August 1991 **Proceedings of the 1991 ACM/IEEE conference on Supercomputing**

Full text available:  [pdf\(1.18 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

20 [Survey of commercial parallel machines](#)



Gowri Ramanathan, Joel Oren

June 1993 **ACM SIGARCH Computer Architecture News**, Volume 21 Issue 3

Full text available: [pdf\(1.64 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [citings](#), [index terms](#)

We have presented in this paper the survey of the parallel machines that are marketed today. The survey includes the latest machines available from Kendall Square Research, Thinking Machines Corporation, MasPar Computer Corporation, NCUBE Corporation, Sequent Computer Systems and Parsytec. We have provided the topology, architecture, cache coherence, synchronization and performance in MFLOPs for each of the machines subject to the availability of information.

Results 1 - 20 of 30

Result page: [1](#) [2](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads: [Adobe Acrobat](#) [QuickTime](#) [Windows Media Player](#) [Real Player](#)

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: The ACM Digital Library The Guide

+invalidate +history +cache +memory +controller +dirty entr

THE ACM DIGITAL LIBRARY

 [Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

[invalidate](#) [history](#) [cache](#) [memory](#) [controller](#) [dirty](#) [entry](#) [line](#) [data](#) [address](#)

Found 30 of 157,956

Sort results by

relevance [Save results to a Binder](#)[Try an Advanced Search](#)

Display results

expanded form [Search Tips](#)[Try this search in The ACM Guide](#) [Open results in a new window](#)

Results 21 - 30 of 30

Result page: [previous](#) [1](#) [2](#)Relevance scale **21 [Hardware-software trade-offs in a direct Rambus implementation of the RAMpage memory hierarchy](#)**

Philip Machanick, Pierre Salverda, Lance Pompe

October 1998 **Proceedings of the eighth international conference on Architectural support for programming languages and operating systems**, Volume 32 , 33 Issue 5 , 11Full text available:  [pdf\(1.47 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The RAMpage memory hierarchy is an alternative to the traditional division between cache and main memory: main memory is moved up a level and DRAM is used as a paging device. The idea behind RAMpage is to reduce hardware complexity, if at the cost of software complexity, with a view to allowing more flexible memory system design. This paper investigates some issues in choosing between RAMpage and a conventional cache architecture, with a view to illustrating trade-offs which can be made in choosing ...

22 [The V distributed system](#)

David Cheriton

March 1988 **Communications of the ACM**, Volume 31 Issue 3Full text available:  [pdf\(2.55 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

The V distributed System was developed at Stanford University as part of a research project to explore issues in distributed systems. Aspects of the design suggest important directions for the design of future operating systems and communication systems.

23 [Principles of transaction-oriented database recovery](#)

Theo Haerder, Andreas Reuter

December 1983 **ACM Computing Surveys (CSUR)**, Volume 15 Issue 4Full text available:  [pdf\(2.48 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#), [review](#)**24 [Functional verification of a multiple-issue, out-of-order, superscalar Alpha processor—the DEC Alpha 21264 microprocessor](#)**

Scott Taylor, Michael Quinn, Darren Brown, Nathan Dohm, Scot Hildebrandt, James Huggins, Carl Ramey

May 1998 **Proceedings of the 35th annual conference on Design automation - Volume 00**

Full text available:  [pdf\(153.68 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)
 [Publisher Site](#)

DIGITAL's Alpha 21264 processor is a highly out-of-order, superpipelined, superscalar implementation of the Alpha architecture, capable of a peak execution rate of six instructions per cycle and a sustainable rate of four per cycle. The 21264 also features a 500 MHz clock speed and a high-bandwidth system interface that channels up to 5.3 Gbytes/second of cache data and 2.6 Gbytes/second of main-memory data into the processor. Simulation-based functional verification was performed on the lo ...

Keywords: 21264, Alpha, architecture, coverage analysis, microprocessor, pseudo-random, validation, verification

25 [I'm done simulating; now what? Verification coverage analysis and correctness checking of the DEC chip 21164 Alpha microprocessor](#)

Michael Katrowitz, Lisa M. Noack

June 1996 **Proceedings of the 33rd annual conference on Design automation**

Full text available:  [pdf\(117.86 KB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



26 [The HP AutoRAID hierarchical storage system](#)

J. Wilkes, R. Golding, C. Staelin, T. Sullivan

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles**, Volume 29 Issue 5

Full text available:  [pdf\(1.60 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)



27 [The HP AutoRAID hierarchical storage system](#)

John Wilkes, Richard Golding, Carl Staelin, Tim Sullivan

February 1996 **ACM Transactions on Computer Systems (TOCS)**, Volume 14 Issue 1

Full text available:  [pdf\(1.82 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Configuring redundant disk arrays is a black art. To configure an array properly, a system administrator must understand the details of both the array and the workload it will support. Incorrect understanding of either, or changes in the workload over time, can lead to poor performance. We present a solution to this problem: a two-level storage hierarchy implemented inside a single disk-array controller. In the upper level of this hierarchy, two copies of active data are stored to provide f ...

Keywords: RAID, disk array, storage hierarchy

28 [A forward-looking method of Cache memory control](#)

J. K. Iliffe

September 1987 **ACM SIGARCH Computer Architecture News**, Volume 15 Issue 4

Full text available:  [pdf\(373.45 KB\)](#) Additional Information: [full citation](#), [index terms](#)



29 [Performance of an OLTP application on symmetry multiprocessor system](#)

Shreekant S. Thakkar, Mark Sweiger

May 1990 **ACM SIGARCH Computer Architecture News , Proceedings of the 17th annual international symposium on Computer Architecture, Volume 18 Issue 3**

Full text available: pdf(1.05 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Sequent's Symmetry Series is a bus-based shared-memory multiprocessor. System performance in an OLTP relational database application was investigated using the TP1 benchmark. System performance was tested with fully-cached benchmarks and with scaled benchmarks. In fully-cached tests, the entire database fits inside main memory. In scaled tests, the database is larger than available memory. In the fully-cached benchmark, performance was initially limited by bus saturation. The cause was the ...

30 The verification of cache coherence protocols

Fong Pong, Michel Dubois

August 1993 **Proceedings of the fifth annual ACM symposium on Parallel algorithms and architectures**

Full text available: pdf(1.05 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Results 21 - 30 of 30

Result page: [previous](#) [1](#) [2](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads: [Adobe Acrobat](#) [QuickTime](#) [Windows Media Player](#) [Real Player](#)

http://portal.acm.org/results.cfm?query=%2Binvalidate%20%2Bhistory%20%2Bcache%20%2B... 7/7/05